



Flexibility Modeling, Management, and Trading in Bottom-up Cellular Energy Systems

Siksnys, Laurynas; Pedersen, Torben Bach; Aftab, Muhammad; Neupane, Bijay

Published in:

Proceedings of the Tenth ACM International Conference on Future Energy Systems, e-Energy 2019

DOI (link to publication from Publisher):

[10.1145/3307772.3328296](https://doi.org/10.1145/3307772.3328296)

Creative Commons License

Unspecified

Publication date:

2019

Document Version

Publisher's PDF, also known as Version of record

[Link to publication from Aalborg University](#)

Citation for published version (APA):

Siksnys, L., Pedersen, T. B., Aftab, M., & Neupane, B. (2019). Flexibility Modeling, Management, and Trading in Bottom-up Cellular Energy Systems. In *Proceedings of the Tenth ACM International Conference on Future Energy Systems, e-Energy 2019* (pp. 170-180). Association for Computing Machinery.
<https://doi.org/10.1145/3307772.3328296>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy

If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Flexibility Modeling, Management, and Trading in Bottom-up Cellular Energy Systems

Laurynas Šikšnys
Aalborg University
siksny@cs.aau.dk

Muhammad Aftab
Aalborg University
muhaftab@cs.aau.dk

Torben Bach Pedersen
Aalborg University
tbp@cs.aau.dk

Bijay Neupane
Aalborg University
bn21@cs.aau.dk

ABSTRACT

Accelerated local deployments of renewable energy sources and energy storage units, as well as increased overall flexibility in local demand and supply through active user involvement and smart energy solutions, open up new opportunities (e.g., self-sufficiency and CO₂ neutrality through local renewables) and yet pose new challenges (e.g., how to maintain the security of supply and get the best yield) to market players in the lower parts of the energy system (including *prosumers*, *energy communities*, *aggregators*, and *distribution system operators (DSOs)*). One way to cope with the challenges requires "logical" reorganization of the energy system *bottom-up* as a number of nested (maximally) self-sufficient and interacting *cells* with their own local (i.e. within a cell) energy management and trading capabilities. This change necessitates effective IT-based solutions. Towards this goal, we propose a unified Flexibility Modeling, Management, and Trading System (FMTS) that generalizes flexibility modeling, management, and intra-cell trading in such cellular energy systems. Our system offers different flexibility provisioning options (Machine Learning based, and Model Predictive Control based), activation mechanisms (indirect and direct device-control), and trading schemes (e.g. flexibility contracts, market-based trading) and suits different cellular system use-cases. In this paper, we introduce the FMTS, overview its core functionality and components, and explain how it practically manages, prices, and trades flexibility from a diverse variety of loads. We then introduce the real-world FMTS instances developed in the GOFLEX project¹ and present experimental results that demonstrate significantly increased flexibility capacities, user gains, and balance between demand and supply when an FMTS instance is used in the simulated cellular energy system setting.

KEYWORDS

Cellular energy systems; FlexOffers; Energy flexibility

¹www.goflex-project.eu

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
e-Energy '19, June 25–28, 2019, Phoenix, AZ, USA
© 2019 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-6671-7/19/06.
<https://doi.org/10.1145/3307772.3328296>

1 INTRODUCTION

More and more renewable energy sources (RESs), energy storage units (batteries), and advanced power metering and energy management solutions are being deployed at the lower levels of the energy system. This opens up new opportunities at the lower levels of the energy system to existing and emerging market players (e.g., prosumers, energy communities, aggregators). These players can finally form (virtual or physical) *micro-grids* by owning, controlling, and optimizing their local electricity assets to best suit their own (and not supplier or system operator) needs of, e.g., self-sufficiency, CO₂ neutrality, or increased utilization of the local renewables at reduced overall cost. This, however, changes the way the traditional (top-down) energy system works and poses new challenges (e.g., intermittent RES production) for local system operators (e.g., DSOs) to control electricity flows in their local grids, the security of which they were traditionally responsible for. To cope with this challenge, the idea of a so-called *cellular energy system* [16, 20, 40] has emerged as a new way of organizing and structuring the lower levels of the existing energy system e.g., (sub-) DSO grid (see Figure 1a). Similar to the Internet, cellular energy system consists of a number of *cells* (sub-networks), where each is defined by a set of players (roles) operating in the area with their own energy assets and energy management and trading/exchange capabilities. Cells are organized hierarchically such that cells may reside within other cells in a manner similar to nesting dolls (see Figure 1a). The cellular system works bottom-up: the smallest cells (sub-systems) are commercial and residential buildings, houses and industrial plants (e.g. performing as micro-grids). The next level corresponds to the energy communities managing energy within a local area containing a number of independent or nested cells. The third level typically corresponds to distribution grids and the fourth level to transmission grids. At each level, market players such as *energy producers*, *energy community operators*, *aggregators*, and *DSOs* are interlinked and have specific responsibilities (and opportunities).

Typically, each cell has to maintain a balance between electricity demand and supply, and account for any imbalances generated (like in the higher levels of the traditional energy system). Maintaining such a balance might be quite challenging at the lower level cells, where significant production typically comes from RES. As RES production is highly volatile, intermittent, and irregular, *flexibility* [7, 10, 17, 21, 23, 24, 28, 30, 38] in demand and supply is essential to counteract. There exist many sources of demand flexibility, e.g., electric vehicles (EVs) or a single heat-pump with its own intrinsic flexibility in time and energy amount [25, 35], as well as sources of

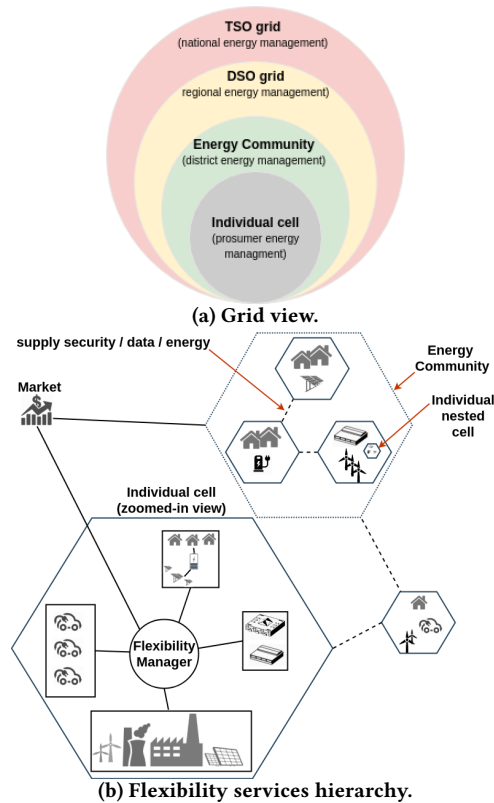


Figure 1: Bottom-up cellular energy system.

supply flexibility, e.g., batteries, EVs, traditional generators. Typically, individual cells may procure and use flexibility for their own needs (which we denote as *self-balancing*). When it is more profitable, the cells may be willing to offer excess flexibility to their neighboring and/or top-level cells (e.g., that of a DSO) to help them balance the grid. Figure 1b describes a bottom-up energy system in terms of such a desired flexibility exchange hierarchy.

Such bottom-up cellular energy systems entail a number of high-level requirements for (the IT infrastructure of) flexibility management, which we group and formulate below.

- (1) **Extensible IT Infrastructure** Individual actors within a cell should be equipped with a flexible and extensible IT infrastructure to manage energy demand and supply flexibilities. It should offer robust inter-operable flexibility provisioning, management, optimization, and trading tools, which can be extended and customized for specific actor needs and their individual business models.
- (2) **Flexibility Management and Exchange** Cell actors need to be equipped with tools to automatically provision, (dis-)aggregate, and optimize flexibility from a diverse variety of unit loads (e.g., EVs), and activate it at desired time, place, and amount in a scalable and timely manner (before physical generation and consumption takes place). As flexibility exchange between cells is desired, the tools have to understand (model) flexibility in a unified and unambiguous way.
- (3) **Flexibility Pricing and Trading** Cell actors need to be equipped with flexibility trading tools that can (1) offer rigorous

pricing schemes for flexibility and its activation, (2) support bi-lateral (flexibility) contracts between actors and actor billing, (3) offer flexibility trading between cells either bi-laterally or via a market system, (4) support traditional electricity markets and power exchanges.

This paper presents a Flexibility Modeling, Management, and Trading System (FMTS) that fulfills these requirements and thus generalizes flexibility modeling, management, and intra-cell trading in such cellular energy systems. In general, FMTS equips cell actors (including prosumers, energy communities, aggregators, DSOs) with an inter-operable stack of IT tools for extracting, (dis-)aggregating, optimizing, pricing/billing generalized flexibility from a variety of loads (unit or aggregated) and allow exchanging flexibility across cell boundaries either bi-laterally and/or via trading in a local flexibility market. FMTS is novel in terms of the unique interplay of concepts, techniques, and components, the majority of which have already been proposed and demonstrated in isolation in (mostly our) earlier work. The key FMTS concept is a so-called FlexOffer [1, 2, 9, 22] that generalizes flexibility from a variety of loads. The three instances of FMTS have been practically deployed and are currently undergoing real-world trials in three European cities. This paper takes a holistic view and aims at presenting the full breadth of this system, primarily focusing on system architecture and integration aspects, and giving references for detailed algorithms. The major contributions of this paper include:

- (1) Showing how FMTS facilitates the cellular energy systems: presenting the conceptual FMTS architecture and its sub-systems and explaining how all sub-systems work and should be used by different actors.
- (2) Explaining how the existing concepts and techniques can work together as a whole in practice: how flexibility portfolios (Flex-Offers) are generated, priced, (dis-)aggregated, optimized, and executed using traditional data management, machine learning, optimization, and specialized techniques.
- (3) Discussing the three pilot instances (demonstrations) of FMTS currently taking place in the GOFLEX project.
- (4) Presenting and discussing the results of a large-scale simulation (replicating the demo sites), which demonstrates that FMTS incurs significant prosumer gains (savings) and yields substantial reductions of imbalances in scenarios with peer and nested cells managing typical consumption/generation flexibility capacities.

The paper is organized as follows. We introduce the FlexOffer concept in Section 2. Section 3 provides an overview of the system architecture. Techniques for flexibility extraction and management is presented in Section 4. We discuss flexibility pricing and trading in Section 5. Pilots and simulated experiments are presented in Section 6 and Section 7. Finally, we conclude and discuss future work in Section 8.

2 THE FLEXOFFER CONCEPT

FMTS relies on the so-called *FlexOffer* (FO), which offers a common unified representation of flexibility in electricity demand and supply. Its initial variant was proposed by the European project MIRABEL [1, 9, 22] and further refined in the subsequent projects

and studies [2, 3, 39]. An FO is a data object which explicitly captures flexibility as a pre-defined set of common constraints (e.g., load activation time and amount). This makes it practical to exchange flexibility information between different entities. For example, in FMTS, FOs are extracted from individual consumption or production flexible resources, e.g., heat pumps, and then propagated through a number of FO manager sub-systems of different cells in the aggregated form, still in the same common FO representation. To cover a wide range of flexible resources, an FO identifies inherent dimensions (types) of flexibility and uses common specialized constraints to characterize each of them. In general, an FO may include the following types of constraints (mathematical inequalities):

- **Start time constraint** – a range defining the earliest and latest start time of consumption or production of a flexible resource.
- **Energy amount constraint** – a range defining the minimum and maximum energy amounts in a given time slice (typically 15 min). An FO typically defines a sequence of such constraints.
- **Total energy constraint** – a range defining the minimum and maximum total energy amount within the full active operation of a flexible resource, e.g., the total amount to charge an EV.
- **Dependent energy amount constraint** – for loads with the rebound effect [15] (e.g., heat-pump loads), the dependent energy constraint captures the minimum and maximum energy amounts in a given time slice in dependence to the total energy consumed or produced at preceding time slices of the active device operation [39].
- **Acceptance time constraint** – a parameter that sets the deadline on when an FO receiving cell should acknowledge successful acceptance or rejection of the FO. An FO rejection may occur if, e.g., FO constraints or other metadata are invalid or inappropriate (e.g., quantities are too small, prices are too high).
- **Assignment time constraint** – a parameter that sets the deadlines on when an FO schedule update (assignment) is allowed to be sent by the FO receiving cell to FO issuing cell. A deadline can be an absolute timestamp or a relative duration with respect to the scheduled operation activation time.

Here, start time and energy amount constraints are often sufficient to capture the flexibility of the most common load types. Given these basic constraints, we define the FlexOffer as follows:

Definition 1. A FlexOffer f is a tuple $f = ([t_{es}, t_{ls}], p, t_{ds}, p_{ds})$, where $[t_{es}, t_{ls}]$ is a time interval defining the *earliest start time* (EST) and the *latest start time* (LST), in which the device operation has to start. p is the energy profile given as a sequence of slices $\langle s_1, \dots, s_d \rangle$, where a slice s_i is a continuous range $[e_{min}^i, e_{max}^i]$ defined by the minimum e_{min}^i and maximum e_{max}^i energy bounds, and d is the number of slices in p . t_{ds} and p_{ds} constitute a so-called *default schedule*: t_{ds} is a time stamp, $t_{es} \leq t_{ds} \leq t_{ls}$, at which the device is expected to start by default (e.g., when not overridden by an external system) and $p_{ds} = \langle e_{ds}^1, \dots, e_{ds}^d \rangle$, $\forall i = 1 : d: e_{min}^i \leq e_{ds}^i \leq e_{max}^i$, are corresponding energy amounts of this operation.

Figure 2 shows an example of an FO f representing the energy demand for a single charging event of an EV. The FO in the figure states that the EV could be charged starting anytime between 4 PM and 12 AM. The FO has an energy profile with five consecutive slices: $s_1 = [0.6, 2]$, $s_2 = [1, 3]$, $s_3 = [0.9, 2.7]$, $s_4 = [0.8, 2.6]$,

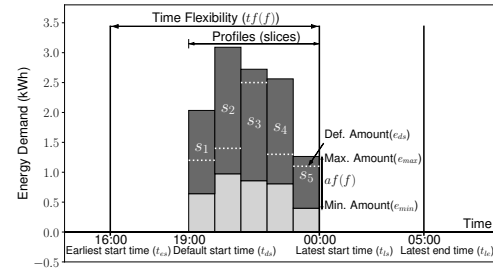


Figure 2: A sample FlexOffer from EV.

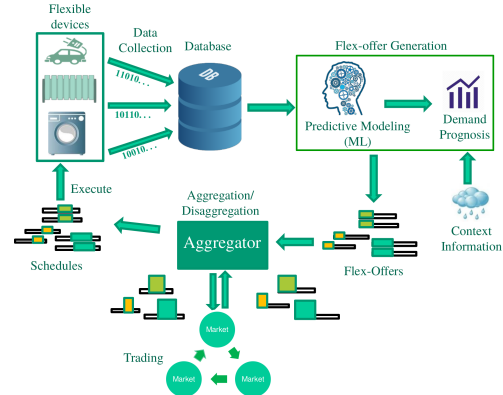


Figure 3: FlexOffer life-cycle – process view.

and $s_5 = [0.4, 1.2]$, where the first and the second elements represent the height of the light-shaded (e_{min}) and dark-shaded (e_{max}) bars in the figure, respectively. The start time flexibility of the FO f is the difference between LST and EST, i.e., $tf(f) = t_{ls} - t_{es} = 8$ hours. Further, t_{le} represents the *latest end time* (LET) of the last slice and is calculated as $t_{le} = t_{ls} + d$. Similarly, the energy amount flexibility $af(f)$ is the difference between max and min energy for each slice, and total energy flexibility is given implicitly by the sum of the differences between amount bound of all slices, i.e., $\sum_{i=1}^d e_{(max,i)} - e_{(min,i)} = 5.8$ kWh. The values captured by an FO largely depend on the flexibility source. For example, FOs from EV charging stations and factories are "large enough", and could be directly traded on the market, whereas FOs from heat pumps and EVs typically represent small flexible loads and need to be aggregated into larger (macro) FOs before trading.

2.1 FlexOffer Life Cycle

From generation until execution, an FO goes through several phases during its life-cycle as depicted in Figure 3 and described below:

- (1) **Data collection:** The first phase involves the collection and storage of energy consumption data from flexible devices (e.g., washing machine, dishwasher, EV, etc.). IoT infrastructure, energy management systems, and off-the-shelf smart plugs may be used to collect data at device level, whereas smart energy meters may be deployed to collect data at house/building level.
- (2) **Predictive model building and demand prognosis:** An FO is essentially an estimation of a load's future behavior and its flexibility, which requires forecasting of the future energy demand and the associated flexibility. Thus, we need to develop

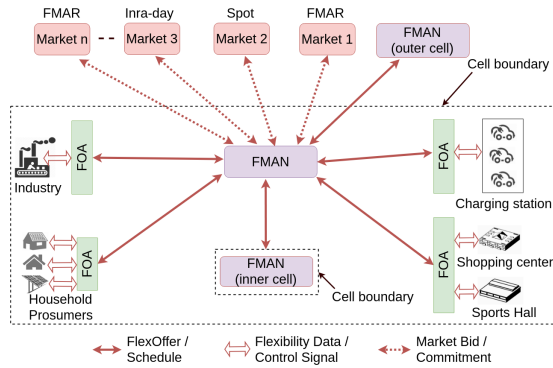


Figure 4: FMTS architecture.

models to analyze user behavior and predict future energy demand [13, 26]. The predicted energy demand and the associated flexibility data is used to generate FOs [27].

- (3) **FlexOffer aggregation/disaggregation:** Individual FOs are combined into large aggregated FOs to make them more useful, valuable, and computationally tractable [31, 32].
- (4) **FlexOffer optimization/trading:** Aggregated FOs are used in direct energy optimization and/or traded on the market.
- (5) **Schedule execution:** In response to aggregated FOs, aggregated schedules are generated. Then, these are disaggregated and forwarded to the corresponding loads.

We explain how FMTS facilitates these phases in Sections 4-5.

3 FMTS OVERVIEW

FMTS is a decentralized system that realizes flexibility (FO) collection, management, and trading, while encompassing all active cellular energy system players, including flexibility providers (prosumers) and flexibility buyers (e.g., energy community aggregators, DSOs). In this section, we overview FMTS core functionality, describing the core components/subsystems and how they interact.

The overall architecture of FMTS is shown in Figure 4. So-called FlexOffer Agents (FOAs) monitor and control physical loads in (near) real-time and offer load-type specific flexibility extraction tools to generate FOs. So-called FlexOffer Managers (FMANs) process and continuously optimize such FOs to meet some desired objective, e.g., minimize cost or make a cell maximally self-sufficient. This objective is set and can be changed over time depending on the business needs of a cell-managing actor (e.g., a prosumer). Further, FMANs also enable inter-cell flexibility transactions. In the nested configuration, an FMAN exchanges flexibility based on bilateral contracts between FMAN and/or FOA users of the inner and outer cells. Flexibility trading between (peer) cells at the same level is handled through a so-called FlexOffer Market (FMAR). FMAR is to be hosted by a third-party or the actor of the outer level cell (containing the two cells). For larger cells (and large flexibility capacities), the FlexOffer Manager can be inter-connected with traditional electricity markets (e.g., local, national, or regional electricity markets like NordPool²), where flexibility can be converted into products for day-ahead, intra-day, or intra-hour trading. Next, we describe the three core sub-systems of FMTS in more detail.

²<https://www.nordpoolgroup.com/>

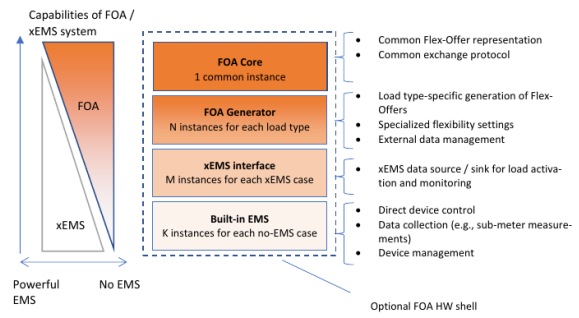


Figure 5: FOA capabilities and internal architecture.

3.1 FlexOffer Agent (FOA)

FlexOffer Agent (FOA) is an extensible and highly customizable component of an FMTS that is responsible for the generation and execution of FOs for one or multiple flexible loads (both production and consumption). It forms individual FOs, delivers FOs to the FMAN, receives disaggregated schedules from the FMAN, and activates the flexible loads according to the received schedules. The generic conceptual FOA architecture is given in Figure 5.

In general, FOA is a component that emits FOs and consumes FO schedules, and acts as a bridge between an FMAN and an energy management system (xEMS) handling multiple physical loads. Therefore, as seen in Figure 5, the actual capabilities of the FOA depend on the available functionality of the xEMS. For example, the FOA serves only as an FO/schedule gateway (interface) when the xEMS is an advanced extensible system with built-in energy prediction and optimization capabilities. When no xEMS is available on site, the FOA then takes the role of xEMS and integrates advanced predictive logic and also performs the function of xEMS (using built-in EMS). In its full configuration, the FOA has four major components, i) core component with all common functionality, ii) generator component for provisioning FOs, iii) xEMS interface for communicating with an existing xEMS, and iv) built-in EMS to control the load if there is no existing EMS. It may also offer a Graphical User Interface (GUI) for administrators (aggregators) and load owners/users for configuration of load and flexibility parameters. The FOA may be deployed either locally at the site of a single load or in a cloud, e.g., for controlling 1000s of loads.

3.2 FlexOffer Manager (FMAN)

The FlexOffer Manager (FMAN) is a central component of a cell. It manages available flexibility, performs energy optimization (e.g., demand-supply balancing), and exchanges FOs either within nested cells (via FMAN of another cell) or trades FOs with other peer cells (via FMAR). It integrates advanced FO aggregation, disaggregation, optimization functionalities, and may offer a GUI, which allows its users analyzing, trading, and shaping available flexibility in near real-time. In simple cases, the FMAN may operate a cell in isolation and offer efficient unit FO aggregation and energy optimization functionalities, e.g., for demand-supply balancing, congestion management. In more advanced cases, the FMAN may first aggregate flexibility (FOs) from FOAs and FMANs at lower levels and then trade the flexibility with other independent adjacent cells through the FMAR. The FMAN is also responsible for (1) disaggregation of the schedules received from the FMAR or FMAN at the upper

level and (2) schedule forwarding to the respective FOAs (or lower level FMANs) for subsequent handling and execution. Finally, the FMAN handles contracts, flexibility prices, and billing based on bilateral contracts between the FMAN user and flexibility providers, as detailed in Section 5.

3.3 FlexOffer Market (FMAR)

FlexOffer Market (FMAR) is a component that realizes the local flexibility market. It can be run by a third party or an outer level cell. The FMAR facilitates flexibility trading and exchange between a number of peer cells belonging to the same shared outer cells. It natively supports flexibility products defined by FOs. Further, the FMAR offers various flexibility trading options based on, e.g., single-sided pool (for cases when there is a single buyer of flexibility) or multi-sided pool (for cases when there are multiple buyers of flexibility). Based on the configuration, the FMAR automatically matches and selects consumption and production FOs from the pool of FOs received from FMANs of various cells and then generates assignments (schedules) for the winning parties. FOs are selected based on the location of the flexible loads, size, price, and its potential for catering the market requirements (e.g., avoiding congestion) [37]. The FMAR is also responsible for forwarding FOs acceptance status and schedules to the respective FMAN, which is eventually forwarded to individual loads for execution. If FO schedules are executed correctly, the involved parties are remunerated based on respective market transactions.

4 FLEXIBILITY EXTRACTION AND MANAGEMENT

4.1 Flexibility Extraction

Generating meaningful and accurate FOs is not trivial. For loads that continuously consume energy (e.g., heat pumps), FOs are typically generated on an hourly or daily basis, whereas other flexible loads (e.g., dishwasher, washer dryer) can emit FOs when needed [25]. Generating FO for a flexible load implies predicting the consumption (or production) of the load while satisfying a given set of constraints (e.g., comfort temperature interval, load capacity). This is most often done using a model of the flexible load, the relevant parameters for the predictions (e.g., temperature, solar radiation), and environmental and other constraints (e.g., comfort settings). The model is employed by FOAs to estimate energy and time flexibility using various techniques as described next.

Human in the loop FO extraction: Human in the loop is the most basic method of FO extraction, where users are asked to interact with the system and provide flexibility parameters. For example, a user can use a dedicated application (a part of FOA) to signal the readiness of the dishwasher or washer dryer for operation. Receiving the signal, the FO generator utilizes the user configured parameters to generate FO for the load operation. This approach is quite accurate but suffers from response fatigue with the users' response rate decreasing gradually over time.

ML-based techniques: Machine Learning (ML) based FOs generation relies on user behavior data and predictive models for estimating various FO parameters, which reduces or altogether eliminates user interaction [18, 27]. The FO generation process starts with the gathering of the energy demand time series and available

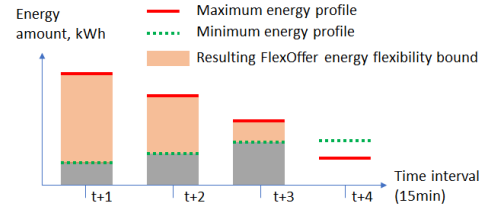


Figure 6: Examples of energy profiles used to compute simple FO energy amount bounds.

context information relevant to the flexible load. For example, to generate an FO for a heat pump, information about occupancy, house insulation parameters, etc., could be collected. The raw information is then preprocessed into a format required for analyzing and predicting timestamps and values for the FO attributes. After preprocessing, the first step is the time flexibility extraction which involves the prediction of timestamps of various actions involved in the operation of flexible resources (e.g., plug-in and plug-out time for EVs). The second step is the amount flexibility extraction which includes the prediction of the number of slices in FOs, i.e., the number of time units required for the completion of the defined task and the minimum and maximum energy bounds for the individual slices. For example, estimating the state of charge (SOC) level for EV. The final step combines the outputs of the previous two steps to generate FOs for the flexible load's future operations.

Dynamic system modeling techniques Dynamic system modeling offers more rigorous techniques to generate FOs. For this, an FOA implements a typical model predictive control (MPC) scheme to control an underlying physical load (or process) in real-time. On top of that, FOA uses the dynamic system model, e.g., *state-space model*, with all underlying process constraints natively used by an MPC controller to estimate available flexibility bounds for inclusion into an FO. Actual bound estimation technique depends on whether simple *energy amount constraint* or *dependent energy constraint* is required (see Section 2).

Energy amount constraint: If simple energy amount constraint is enabled, FOA periodically (e.g., every 15 mins) solves an optimization problem, which predicts future energy loads (profiles) under maximum and minimum power conditions assumed for the processes in the present state at some time t . As shown in Figure 6, FO energy amount bounds of the consecutive slices correspond to maximum and minimum predicted energy amounts of the two profiles – but only those with valid bounds ($t + 1..t + 3$).

Dependent energy constraint: If more advanced dependent energy constraint is enabled, FOA periodically (e.g., every 15 mins) makes a symbolic rewriting [39] of the model constraints by applying Fourier-Motzkin variable elimination [11] to produce a system of equations in a format required by this FO constraint. This technique allows preserving inherent energy dependencies between consecutive time intervals (e.g., of a rebound effect [15]) while still offering scalable FO aggregation (and optimization) [39].

4.2 Aggregation/disaggregation

FOs from individual flexible resources (e.g., heat pumps, EVs) typically represent small flexible loads. Thus, a single (small) FO has low impact and is of little interest for peak shaving, electricity trading, and balancing demand and supply in a cell, where required

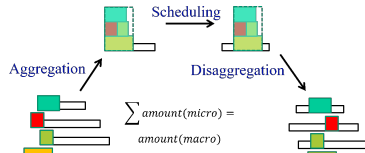


Figure 7: FlexOffer aggregation and disaggregation [29].

balancing capacities are much higher. At the same time, managing large numbers of individual FOs is tedious and complex. A common solution is to utilize FO aggregation, where flexibilities from individual flexible loads are combined and offered in a more useful and effective aggregated form [4, 12, 32–34, 37, 39]. The aggregated FOs have much larger energy amounts and flexibility margins and are easier to manage. Aggregating large number of FOs, however, is a computationally hard problem, which requires dealing with many decision variables and constraints originating from many FOs.

To optimize aggregation, FOs can be grouped based on a similarity measure (e.g., consumption pattern). Aggregation is typically performed by entities called aggregators using FMANs. An FMAN receives FOs from individual FOAs (or lower level FMANs) and then aggregate these FOs. The flexibility of aggregated FOs tends to be lower than the joint flexibility of the FOs that compose them. This reduction in flexibility is, however, unavoidable in order to reduce FOs scheduling complexity and to increase their value (e.g., on the flexibility market). After aggregation, schedules are typically assigned to the aggregated FOs (e.g., based on energy sold on the market). By respecting all inherent aggregated FOs constraints, a schedule specifies the exact start times and aggregated energy amounts assigned to the underlying flexible resources. Such schedules are disaggregated to schedules for each individual FO it is composed of. This operation is denoted FO disaggregation. Disaggregated schedules are finally forwarded to the flexible resources which initially offered flexibility. The process of FO aggregation, scheduling, and disaggregation is illustrated in Figure 7 and explained in greater detail in [32].

4.3 Optimization

Common shared representation of flexibility (FOs) enables highly robust energy optimization techniques inside FMTS, which are tailored specifically for FOs. As such, FMTS dynamically chooses an actual solving technique depending on (1) optimization objective (e.g., demand-supply balancing, portfolio cost reduction, energy maximization/minimization within a period), and (2) types of constraints enabled and used inside FOs. For example, FMTS uses standard *linear programming* techniques to generate schedules for simple linear objectives and basic *energy amount*, *total energy constraint*, and *dependent energy amount* constraints. When *start-time constraint* is used inside FOs, FMTS uses *mixed-integer programming* (MIP) to find (semi-)optimal schedules with *start-time* parameters taking discrete values. Alternatively, the best effort techniques like *simulated annealing*, *hill-climber* can be used to cope with such discrete-value constraints. For more complex planning objectives and various combinations of FO constraints, a specialized genetics-inspired technique [33] is used. In all cases, FO aggregation is performed prior to optimization to reduce the total number of decision variables and thus the overall planning complexity. This enables our optimization techniques to converge much faster than

the 15-minute planning period standardized in FMTS. A detailed discussion on the optimization result quality and the execution time is provided in [33].

4.4 Scheduling

Once an aggregated FO has been optimized and a schedule assigned to it, the FMAN disaggregates the schedule of the aggregated FO into their respective individual schedules denoting the exact time and amount of energy that has to be consumed by each individual load. The FOA is responsible for executing the received schedule. For each received schedule, the FOA first verifies that the schedule satisfies the constraints encoded in the FO. If the schedule is invalid, the default schedule is executed, and the issuing component (FMAN or FMAR) is notified about the event. In case of valid schedule, the FOA operates the flexible load according to the received schedule. Depending on the unit type, the schedule can either be stored on the unit itself or in the cloud instance of FOA.

4.5 Monitoring and Control

The FOAs of FMTS support energy management for various types of end users (factory, household, EV charging station center). It basically provides the following two services 1) monitoring of individual flexible load operation and 2) control of the individual flexible load operation according to the needs of both the prosumer and the flexibility end user (e.g., aggregator). The monitoring includes tasks such as checking health, recording energy consumption, efficiency, etc., whereas control includes tasks such as the execution of a schedule and maintaining the normal operation. Often, the monitoring and control of a flexible load are performed via a third-party xEMS system, as discussed in Section 3.1.

5 FLEXIBILITY TRADING AND PRICING

Currently, grid actors trade electricity on existing traditional day-ahead (spot), intra-day, and regulation energy markets [36]. Our system also utilizes a so-called flexibility market (i.e., FMAR). It is based on a variation of the product-mix auction [19], in which the commodities are flexible energy loads for specific geographical areas. This model is designed to deal with the product mix problem, in which multiple varieties of a product with different costs are supplied, but with a constraint on the total capacity. Here the product is flexibility, and the varieties are positive and negative flexibility. Each bidder can make one or more bids, and each bid contains a set of mutually exclusive offers. Bids in the flexibility market are in fact mutually exclusive, since using both negative and positive flexibility for a given geographical area would not make sense. Two types of bids can be made, supply bids, offering flexibility, and demand bids requesting it. The auctioneer then selects the market clearing price that for each bid gives bidders the greatest surplus. Offers with a negative surplus are rejected. This is visualized in Figure 8. In both graphs, a bid is represented by a horizontal segment. The length of a segment determines the supplied or demanded flexibility amounts, while its position on the Y-axis shows the associated price. On the left side, Up Bids correspond to bids for negative flexibility, while Down Bids are for positive flexibility. The market clearing price is found at the intersection between demand and supply lines. Supply bids above this line are rejected, similar to demand.

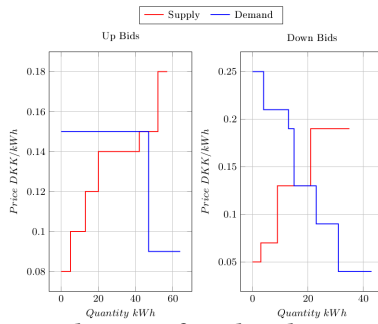


Figure 8: Visualization of market clearing price [22].

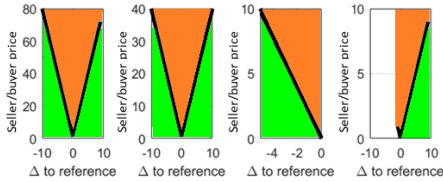


Figure 9: Pricing of flexibility based on FO contracts. Green and orange areas represent potential buying and selling prices, respectively.

FMTS supports two trading modes: delegated and direct:

Delegated Trading: A cell may exchange its flexibility with another cell (e.g., of aggregator), which will reward the cells based on a pre-agreed bi-lateral contract. Such contract includes multiple parameters, which rigorously define how flexibility should be priced. The cells will only report their flexibility (through FOA or FMAN), and the actual flexibility price will be calculated by the FMAN of the delegated cell. This trading mode is to be used by smaller cells (e.g., of prosumer) with no support for local energy optimization.

Direct Trading: A cell is allowed to actively price its offered flexibility (through its FOA or an FMAN) before exchanging it with FMAN. FMAN may decide to reject the offer, e.g. if the price is too high. This trading mode is to be used by medium and large cells (e.g., prosumers, aggregators) capable of actively performing local energy optimization while taking actual energy prices into account. In direct trading, the price of flexibility is incorporated into an FO in the form of “V-shaped price curves”, described next.

5.1 Flexibility Pricing

As given in Definition 1, an FO includes a *default schedule* (aka., baseline) which expresses the flexibility provider’s (e.g., prosumer’s) preferred (often locally optimal) energy amounts to be produced or consumed in the future ($e_{ds}^1, \dots, e_{ds}^d$), along with energy bounds ($[e_{min}^1, e_{max}^1], \dots, [e_{min}^d, e_{max}^d]$) in which this default schedule can be varied (overridden) by a flexibility buyer (e.g., aggregator). To reward the provider for implementing this variation, each FO may define, a so-called *deviation price*, which is a minimum price to be paid for requesting a 1 kWh deviation from the default schedule (baseline). Within the available energy bounds, this yield d V-shaped price curves (in the linear case) associated with each discrete time interval of active device operation. – see Figure 9. Note, this deviation price also allows defining maximum buying price in the case of a buying FO bid issued on the market.

The price not smaller than the deviation price is paid for the successful activation of flexibility (a deviation from the baseline).

Thus, flexibility can be dynamically priced based on parameters such as offered time interval, grid location, flexibility type, etc. Alternatively, in the delegated trading case, flexibility may be treated as an asset and priced in the explicit form (e.g., monthly), where the flexibility buyer may reward the flexibility provider based on the number of FOs, total time and energy flexibility, etc. offered.

6 LARGE-SCALE PILOTS AND SIMULATION

6.1 Real-world Pilots

The system is deployed and being tested at three different European pilot sites, where each individual pilot case offers different aspects of the electricity systems, aiming to include every reasonably encountered prosumer and process. In all pilot cases, FMTS is instantiated at its full scale with a number of FOA instances controlling different type of loads (industrial loads, household loads, and EVs), a number of aggregator instances based on FMAN, and a local flexibility market based on FMAR. Below, the FMTS configuration is given for each pilot case.

6.1.1 German Pilot Case. The first pilot site is in the city of Wunsiedel, Germany, which aims to become self-sufficient by synchronizing and decentralizing energy production and distribution. The utility company SWW plays the role of both energy provider and DSO. The pilot case aims to involve all SWW prosumers with PV production, 4 wind turbines, and several power plants, with the objective to minimize corrective costs and losses for SWW and reduce peak loads by utilizing the FMTS solution.

6.1.2 Swiss Pilot Case. The second pilot site is in Valais, Switzerland. The utility company ESR serves as both energy provider and DSO. The pilot case involves over 200 prosumers with PV installations and flexible loads (such as boilers, heat pumps, etc.). The pilot objectives include the reduction of corrective costs for ESR (e.g., using one day-ahead planning to reduce intra-day correction costs), and using DSM to reduce peak loads.

6.1.3 Cyprus Pilot Case. The third pilot site is located at the University of Cyprus in Nicosia. The pilot case includes various campus facilities, PV systems, and storage systems, aiming to emulate different prosumer types and market activities. As in the previous pilot cases, the main focus of this pilot case is utilizing the FMTS solution to maximize self-consumption through intelligent shaping of daily consumption profiles according to distributed RES generation.

6.1.4 Preliminary Trial Results. The system trials have been going on for seven months as of May 2019. We only consider prosumers with no xEMS loads (i.e., connected to Smart Plugs). So far, the system has 29 such prosumers with 93 flexible loads across the three pilot sites. Presently, 3.95 FOs are generated per load per day, which amounts to 0.78 MWh of flexible energy per month. The ratio of flexible energy to the total energy (from flexible loads) is 11% under a relatively conservative flexibility extraction strategy. When the trials run at full scale, the number of prosumers and flexible loads are expected to exceed 80 and 250, respectively. The total flexible energy is expected to exceed 3 MWh per month. Similarly, the ratio of flexible energy to the total energy is expected to be >20% as a less conservative flexibility extraction strategy is adopted.

6.2 Simulation

As described in the preceding section, a number of FMTS instances have already been deployed in three pilot sites. However, at the time of writing this paper, we have not collected enough data for extensive evaluation. Accordingly, we test FMTS in a simulated environment at a large scale with many users and loads, and under different cell configurations. To simulate all kinds of flexible loads, we have developed a so-called Flexible Load Simulator (FLS), which is a client-server-based simulation tool based on *Modelica* [14]. FMTS sub-systems can connect to FLS to emulate physical loads in a co-simulation mode with a global clock, where the sub-systems can access simulated data and control a number of simulated processes (electric loads) in online fashion. FLS provides the same interface and load control logic as in the pilot sites, thus providing a replica of the actual pilot sites to FMTS for realistic evaluation. Internally, FLS stores the flexible load models as functional mock-up units (FMUs), which are standard inter-changeable units to store the models of physical systems [8]. FLS first instantiates all the models (FMUs of flexible loads) and then starts the real-time simulation of the models. FOAs may post the desired control signal for forcing a change in load behavior or changing model parameters. For simplicity, the simulation engine treats all the system models independently.

For simulation, we use the following two types of load models [6]:

- (1) **Exponential Decay Model:** The power consumption rate of such loads follows an exponential decay curve, dropping from the initial surge power p_{peak} to a stable power p_{active} at a decay rate λ . Refrigerators and Freezers usually follow this model ($p(t) = p_{active} + (p_{peak} - p_{active})e^{-\lambda t}$) [6].
- (2) **Heat Pump Model:** The heat pump system is modeled as an ordinary linear time-invariant (LTI) single-input-single-output (SISO) first order system. Details are provided in [39].

7 RESULTS

In this section, we present the results of the simulated experiments to evaluate different aspects of FMTS (RES utilization, grid balancing, and user gains), when the grid is organized as cells with local consumption and production units and load management.

7.1 Experiment Setup

The experiments have been designed to simulate typical cellular grid scenarios. We involve three cells: C1 and C2, and C12 which includes C1 and C2. Each cell is managed by a dedicated FMAN (hereafter called FMAN1, FMAN2, and FMAN12). C1 and C2 are configured to have 25 household prosumers distributed across the cell area, and each prosumer is set to have 5 flexible loads including two electric heat pumps, one refrigerator, one freezer, and a PV. These loads are chosen because they are common in the pilot sites. Table 1 provides the details of each load. In addition, each prosumer has some inflexible baseload which is derived from the average electricity demand profile in DK1 region of Denmark [5].

We consider three different cell interaction scenarios:

- **Local Cell Optimization:** In this baseline scenario, C1 and C2 performs local demand and supply balancing with the objective to maximize the self-consumption of RES and reduce imbalances towards C12.

Load Name	Model	Parameters
Refrigerator	Exponential Decay	$p_{peak}=650.5\text{Watt}$, $p_{active}=126.2\text{Watt}$, $\lambda=0.27$
Freezer	Exponential Decay	$p_{peak}=430.0\text{Watt}$, $p_{active}=120.1\text{Watt}$, $\lambda=0.25$
Heat Pump	Single Input Single Output (SISO) Linear System	$A=-0.01$, $B=0.002$, $C=1$, $D=0$
Solar PV	Solar PV System	Typical summer day PV production profile in Denmark

Table 1: Simulated loads used in the experiment.

Contract Parameter	Value
Fixed reward for issuing at least 1 FlexOffer	0.03 Euro
Reward of energy flexibility	0.05 Euro/kWh
Reward for a single unit (15 min) of time-flexibility	0.05 Euro

Table 2: General prosumer contract.

- **Peer Cell Trading:** In this scenario, C1 and C2 trade flexibility in a peer-to-peer configuration to mitigate the effect of their imbalances through the FMAR of C12.
- **Nested Cell Trading:** In this scenario, C12 activates C1 and C2 flexibility through bilateral contracts.

The general contract detailed in Table 2 is used to calculate end-prosumer rewards/profits during the experiment in the case of C1 and C2. The values in the contract are estimated based on average European residential customer electricity price of 0.21 Euro/kWh³.

7.2 Local Cell Optimization

First, we demonstrate the effect of local optimization of demand and supply. In this experiment, both C1 and C2 have flexible loads and their respective FOAs generate (micro) FOs using ML-based FO generation techniques discussed in Section 4.1. FMAN1 and FMAN2 independently aggregate the FOs and make them available for optimization (and trading). The optimization is done by FMAN1 and FMAN2 for the objective of maximizing RES self-utilization and minimizing imbalances between local demand and supply. In this regard, we consider C1 with excess RES production at some hours of a day, and C2 with RES production always below the demand. Figures 10-11 show the default (non-optimized) and optimized energy demand of C1 and C2, respectively. The solid lines show the default demand, the dashed lines represent the optimized demand, and the light shaded regions represent supply from RES.

Figure 10 shows that for the particular day, C1 had an excess supply of 97 kWh over a span of 2 hours (region of the plot where RES curve surpasses the default demand curve). The self-optimization of C1 schedules some of the demand from the time period with lower RES production to the time period with excess RES production (dashed line). This results in an increase of 54 kWh in RES self-consumption, which amounts to 38.2 kg reduction in CO₂ emissions⁴. This is mainly achieved by exploiting the demand flexibility from flexible loads, at the cost 4.38 EUR rewarded to prosumers based on the contract values shown in Table 2 (details in Section 7.5). On the other hand, the self-optimization of C2 does not reschedule any energy and the optimized schedule is the same as the default

³<https://1-stromvergleich.com/electricity-prices-europe>

⁴<https://www.epa.gov/energy/greenhouse-gas-equivalencies-calculator>

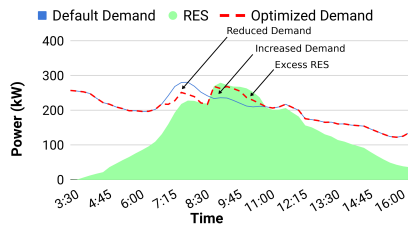


Figure 10: Local demand and supply balancing for Cell 1.

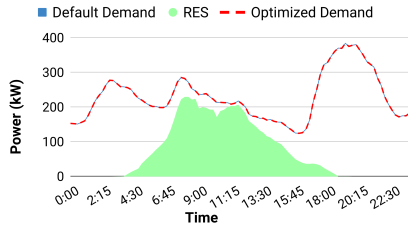


Figure 11: Local demand and supply balancing for Cell 2.

schedule because RES production is always lower than the demand (Figure 11) and no (positive) flexibility is available to counter-act.

In both cases, C1 and C2 are still in a state that lead to substantial imbalances (up to 93 kWh) generated towards C12. Depending on applicable fees (e.g., electricity prices or imbalance fees), these imbalances might be acceptable or not to C1 and C2. However, C1 and C2 have the opportunity to mutually clear out those imbalances and/or generate additional profit for available flexibility through peer-to-peer cell trading.

7.3 Peer-To-Peer Cell Trading

We consider a local peer-to-peer trading scenario, where C1 chooses to ignore (potential) imbalances and willing to compensate this imbalance overhead by selling flexibility (when the expected flexibility price is greater than the imbalance price); and C2 chooses to reduce imbalances directly by acquiring additional supply capacity (when the expected flexibility price is less than the imbalance price). In this scenario, FMAN1 and FMAN2 generate bids for C1 and C2 in the form of FOs, respectively, and forward them to the FMAR of C12 every 15 min. The FOs define locally optimized energy amounts (Figures 10 and 11) and possible deviations along with prices to be paid to/by C1 and C2 for activating these deviations, taking a V-shape form as discussed in Section 5.1. For example, in the period 09:15 to 09:30 in Figure 10, we can see that even after local optimization C1 has 12 kWh of excess RES production. FMAN1 converts the excess supply along with flexibility to further increase RES supply (by reducing demand) to a selling production FO seen in Figure 12a and forwards it for trading on FMAR. Similarly, as seen in Figure 11, C2 is exposed to an imbalance of 14 kWh in this interval, and so FMAN2 generates the buying consumption FO shown in Figure 12b. It specifies the need to reduce demand in C2 together with the maximum deviation price C2 is willing to pay for a deviation of 1 kWh.

FMAR continuously matches the bids and generates new schedules based on the deviation prices and energy amounts of the buying and selling FOs. For example, in this case, the bid FOs shown in Figure 12 are matched at a C1 deviation amount of -2 kWh (=12 kWh - 14 kWh) and the marginal deviation cost of 0.42 EUR paid to C1 by C2. The matched energy amounts become firm execution

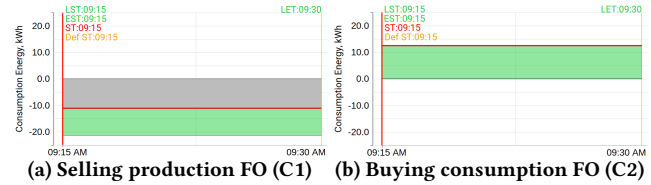


Figure 12: Example of the generated FOs.

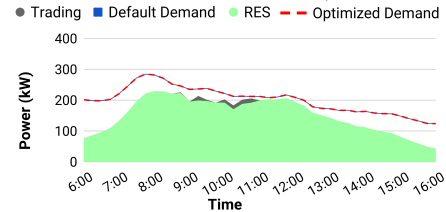


Figure 13: Peer cell trading.

schedules for C1 and C2 and will later be used as references for calculating C1 and C2 imbalance amounts in this time period, while clearing out the imbalances of 14 kWh both for C1 and C2. As result, C1 generates additional profit of 0.42 EUR and, at the same time, spares some demand up-regulation capacity (which can be traded again), while C2 completely eliminates the (potential) imbalance of 14 kWh for this time interval at the marginal cost of 0.42 EUR.

Figure 13 shows the final demand and supply schedules of C2 after executing 36 FMAR trading cycles in the period 6:00 and 15:00. The dark shaded area in the figure represents flexibility traded with C1. Through flexibility trading, C2 is able to decrease its imbalance by 18 kWh (42% of the excess RES from C1), and reduce CO₂ emission by 12.7 kg. Finally, C1 is left with a remaining 25 kWh of flexibility capacity, which could be traded with other cells.

In this scenario, we have considered one flexibility buyer and one seller. However, we note that FMAR also matches two selling (or buying) consumption and production FOs when there is no active flexibility buyer (or seller). This happens at the marginal deviation cost of 0 Eur, and only if FO energy amounts counter-act each other, e.g., one FO defines positive and the other defines negative energy values, like in the case of C1 and C2. In such a case, imbalance amounts are automatically transferred from one participant to another at no incurred deviation cost. Thus, the participants can switch from the role of flexibility buyer to seller (and vice versa) depending on dynamic real-time conditions.

7.4 Nested Cell Trading

Similar to prosumers offering flexibility to aggregators (through FOAs), a cell can delegate management (and trading) of flexibility (through an FMAN) to a higher-level cell in the nested configuration. After an FMAN performs aggregation and local cell balancing, it can automatically send (some of) the aggregated FOs to the FMAN of the higher-level cell. Such aggregated FOs may cover the whole area of the cell or target some specific point of congestion in the grid. Activation of such FOs is automatically remunerated based on the bilateral contract (similar to the one shown in Table 2). Such flexibility delegation gives additional business opportunities to the cells, and may be used, e.g., to provide additional reserve capacity in the nested cell configuration.

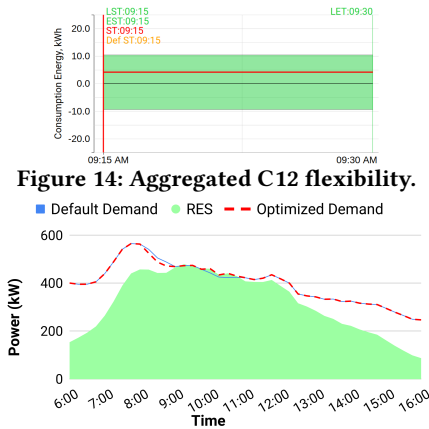


Figure 14: Aggregated C12 flexibility.

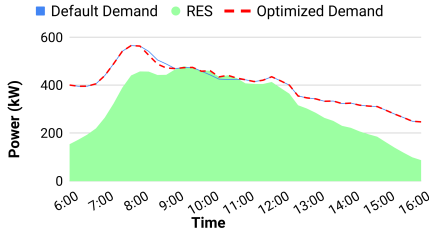


Figure 15: Nested cells.

Let's again consider C1 and C2 residing inside C12. We now run an experiment to compare C12 load balancing through nested cell trading with that through peer-to-peer trading, discussed earlier.

We assume that the FMAN of C12, FMAN12, is allowed to utilize all aggregated C1 and C2 FOs (the total aggregated flexibility in the two areas). Thus, to balance demand and supply, FMAN12 can override optimized schedules of FMAN1 and FMAN2 when needed, expectedly at much higher cost compared to peer-to-peer trading. To reflect that, we assume C1-C12 and C2-C12 contracts use contract parameter values 50% higher than those used for the prosumer contracts (Table 2). Thus, we account for additional flexibility aggregation overhead incurred by C1 and C2.

The results are shown in Figures 14 and 15. Figure 14 shows the overall aggregated flexibility of 10.8 kWh in up-regulation and 9.3 kWh in down-regulation in the period from 09:15 to 09:30 for C12, the combined area of C1 and C2. Note, this flexibility can, potentially, be traded again by C12 with a higher-level cell bilaterally or peer cells in a peer-to-peer fashion. Lastly, Figure 15 shows the overall C12 balancing effect. The dashed line represents 12 kWh of energy re-scheduled by FMAN12 to match the combined RES production from C1 and C2. This yields the full utilization of the RES production in the period 9:00 and 11:00 and a reduction of 68.6 kg in CO₂ emissions. To achieve this balance, FMAN12 has somewhat randomly chosen between relevant C1 and C2 FOs and did not favour any specific cell when activating flexibility. This is because we have used identical C1-C12 and C2-C12 contracts. Setting lower contract values in one automatically favours the FOs of a respective cell (with cheaper flexibility). Overall, trading in this nested configuration resulted in 67% improved C12 balance compared to peer-to-peer trading, but at 90% higher cost. This cost overhead is 27% and 153% when we set the contract values equal to 1x and 2x over the values in Table 2, respectively. All in all, through such bilateral contracts and FMAR bids (parameters of which both can evolve over time), a cell may expose and thus find the best buyer for its operational flexibility.

7.5 Prosumer Gains

Here, we compare incurred end-prosumer rewards for providing flexibility. As a reference, we use the local cell optimization scenario (Section 7.2) and the following different flexibility rewarding schemes: i) only a fixed reward for each generated FO, ii) reward for

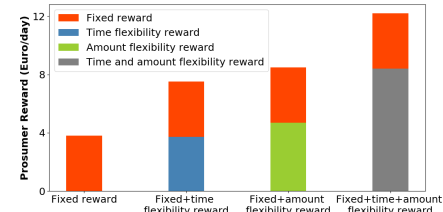


Figure 16: Prosumer utility under different scenarios.

time flexibility only, iii) reward for energy amount flexibility only, iv) reward for both time and amount flexibility. This comparison can be seen in Figure 16.

The figure shows that average daily reward for the number of generated FOs is somewhat comparable to the average daily reward for time and amount flexibility, when the contract values from Table 1 are used. This is mainly because we reward the prosumer regardless of whether or not the generated FO is also activated. Hence, making the reward conditional to FO activation will significantly reduce the rewards under this category. The average daily fixed, time flexibility, amount flexibility, and combined (time and amount) flexibility rewards for a typical prosumer are 3.8, 3.7, 4.8, and 8.4 Euro, respectively. At the same time, this is also an expense of a higher-level cell player (e.g., aggregator).

8 CONCLUSION AND FUTURE WORK

This paper presented Flexibility Modeling, Management, and Trading System (FMTS) that offers an extensible IT solution and supporting tools for extracting, (dis-)aggregating, optimizing, pricing/billing generalized flexibility from a variety of loads in FlexOffer form. The system allows all active flexibility providers, intermediaries, and flexibility users to exchange flexibility between cells either bi-laterally and/or via trading in a local flexibility market.

The paper presented the conceptual architecture of FMTS, its core functionality, its inherent sub-systems, and explained how the sub-systems interact and should be used by different actors. It explained how FlexOffers are generated, (dis-)aggregated, optimized, and executed using traditional machine learning, optimization, and specialized techniques. Then, it presented three pilot demonstrations of FMTS, which are currently taking place in the GOFLEX project. The results of large-scale simulation (replicating the demo sites) demonstrate that FMTS improves RES self-utilization by 55% using self-optimization, which is further improved by 19% when peer cells manage and trade flexibilities. Notably, the system allows for 100% RES self-utilization within the full extent of the imbalance period in the nested cell configuration but at a higher price compared to peer-to-peer trading. In all cases, the system yields significant end-prosumer gains (savings).

Future work will focus on further enriching our techniques for various tasks such as demand prognosis, cell optimization, market trading, etc. An open-source reference implementation of FMTS which includes comprehensive FOA and FMAN implementations accessing the third-party FMAR web service is available online under <https://www.daisy.aau.dk/flexoffers>.

ACKNOWLEDGMENTS

This work was supported in part by the GOFLEX project funded under the Horizon 2020 programme.

REFERENCES

- [1] 2013. The MIRABEL Project, 2013. <http://www.mirabel-project.eu>.
- [2] 2016. The GOFLEX Project, 2017. <https://www.goflex-project.eu>.
- [3] 2016. The TotalFlex Project, 2014. <http://www.totalflex.dk/Forside/>.
- [4] 2017. The Virtual Market of Energy (VME) implementation developed in the Arrowhead project. https://github.com/lawrizs/ARROWHEAD_VME.
- [5] 2019. Nord pool market, 2017. <https://www.nordpoolgroup.com/trading/Day-ahead-trading/Order-types/Flexi-order/>.
- [6] Muhammad Aftab and Chi-Kin Chau. 2017. Smart Power Plugs for Efficient Online Classification and Tracking of Appliance Behavior. In *Proceedings of the 8th Asia-Pacific Workshop on Systems (APSys '17)*.
- [7] Lukas Barth, Veit Hagenmeyer, Nicole Ludwig, and Dorothea Wagner. 2018. How much demand side flexibility do we need?: Analyzing where to exploit flexibility in industrial processes. In *Proceedings of the Ninth International Conference on Future Energy Systems*. 43–62.
- [8] T. Blochwitz, M. Otter, M. Arnold, C. Bausch, H. Elmqvist, A. Junghanns, J. Mauszig, M. Monteiro, T. Neidhold, D. Neumerkel, H. Olsson, J.-V. Peetz, S. Wolf, and C. Clauszig. 2011. In *Proceedings of the 8th International Modelica Conference; March 20th-22nd; Technical University; Dresden; Germany*. 105–114.
- [9] Matthias Boehm, Lars Dannecker, Andreas Doms, Erik Dovgan, Bogdan Filipić, Ulrike Fischer, Wolfgang Lehner, Torben Bach Pedersen, Yoann Pitarch, Laurynas Šikšnys, et al. 2012. Data management in the mirabel smart grid system. In *EDBT/ICDT Workshops*. 95–102.
- [10] Kristof Coninx, Geert Deconinck, and Tom Holvoet. 2018. Who gets my flex? An evolutionary game theory analysis of flexibility market dynamics. *Applied Energy* 218 (2018), 104–113.
- [11] George B Dantzig and B Curtis Eaves. 1973. Fourier-Motzkin elimination and its dual. *Journal of Combinatorial Theory, Series A* 14 (1973), 288 – 297.
- [12] Luis Lino Ferreira, Laurynas Šikšnys, Per Pedersen, Petr Stluka, Christos Chrysoulas, Thibaut Le Guilly, Michele Albano, Arne Skou, César Teixeira, and Torben Pedersen. 2014. Arrowhead compliant virtual market of energy. In *IEEE ETEA*.
- [13] Davide Frazzetto, Bijay Neupane, Torben Bach Pedersen, and Thomas Dyhre Nielsen. 2018. Adaptive User-Oriented Direct Load-Control of Residential Flexible Devices. In *Proceedings of the Ninth International Conference on Future Energy Systems*. 1–11.
- [14] Vadim Fritzon, Peterand Engelson. 1998. Modelica — A unified object-oriented language for system modeling and simulation. In *ECOOP'98 — Object-Oriented Programming*.
- [15] Lorna A. Greening, David L. Greene, and Carmen Difiglio. 2000. Energy efficiency and consumption – the rebound effect – a survey. *Energy Policy* 28 (2000), 389 – 401.
- [16] Hui-Ju Hung, Ting-Yu Ho, Shi-Yong Lee, Chun-Yuan Yang, and De-Nian Yang. 2018. Relay Selection for Heterogeneous Cellular Networks with Renewable Green Energy Sources. *IEEE Transactions on Mobile Computing* 17 (2018), 661–674.
- [17] Rune GrÅynborg Junker, Armin Ghasem Azar, Rui Amaral Lopes, Karen Byskov Lindberg, Glenn Reynnders, Rishi Relan, and Henrik Madsen. 2018. Characterizing the energy flexibility of buildings and districts. *Applied Energy* (2018), 175–182.
- [18] Dalia Kaulakienė, Laurynas Šikšnys, and Yoann Pitarch. 2013. Towards the automated extraction of flexibilities from electricity time series. In *Proceedings of the Joint EDBT/ICDT 2013 Workshops*. ACM, 267–272.
- [19] Paul Klemperer. 2010. The product-mix auction: A new auction design for differentiated goods. *Journal of the European Economic Association* (2010), 526–536.
- [20] Benjamin Kroposki, Emiliano Dall'Anese, Andrey Bernstein, Yingchen Zhang, and Bri-Mathias Hodge. 2018. Autonomous Energy Grids. (2018).
- [21] Merla Kubli, Moritz Looch, and Rolf Wüstenhagen. 2018. The flexible prosumer: Measuring the willingness to co-create distributed flexibility. *Energy Policy* 114 (2018), 540–548.
- [22] Thibaut Le Guilly, Laurynas Šikšnys, Michele Albano, Per Pedersen, Petr Stluka, Luis Lino Ferreira, Arne Skou, Torben Bach Pedersen, and Petur Olsen. 2016. An energy flexibility framework on the internet of things. *The Success of European Projects using New Information and Communication Technologies* (2016), 17–37.
- [23] Anna Joanna Marszał, Jakob Stoustrup, Joakim Widén, and Jérôme Le Dréau. 2017. Simple flexibility factor to facilitate the design of energy-flex-buildings. In *IBPSA*. 503–510.
- [24] Brida V Mbuwir, Mahtab Kaffash, and Geert Deconinck. 2018. Battery Scheduling in a Residential Multi-Carrier Energy System Using Reinforcement Learning. In *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. 1–6.
- [25] Bijay Neupane, Torben Bach Pedersen, and Bo Thieson. 2014. Towards Flexibility Detection in Device-Level Energy Consumption. In *ECML/PKDD Workshop, DARE'14*. 1–16.
- [26] Bijay Neupane, Torben Bach Pedersen, and Bo Thieson. 2018. Utilizing Device-level Demand Forecasting for Flexibility Markets. In *e-Energy '18*. 108–118.
- [27] Bijay Neupane, Laurynas Šikšnys, and Torben Bach Pedersen. 2017. Generation and Evaluation of Flex-Offers from Flexible Electrical Devices. In *e-Energy '17*. 143–156.
- [28] Konstantinos Oikonomou and Masood Parvania. 2019. Optimal coordination of water distribution energy flexibility with power systems operation. *IEEE Transactions on Smart Grid* 10, 1 (2019), 1101–1110.
- [29] T. B. Pedersen, L. Šikšnys, and B. Neupane. 2018. Modeling and Managing Energy Flexibility Using FlexOffers. In *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. 1–7. <https://doi.org/10.1109/SmartGridComm.2018.8587605>
- [30] Joakim Børhum Petersen, Jan Dimon Bendtsen, Pierre Vogler-Finck, and Jakob Stoustrup. 2018. Energy Flexibility for Systems with large Thermal Masses with Applications to Shopping Centers. In *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. 1–6.
- [31] Laurynas Šikšnys, Mohamed E. Khalefa, and Torben Bach Pedersen. 2012. Aggregating and Disaggregating Flexibility Objects. In *SSDBM*. 379–396.
- [32] Laurynas Šikšnys, Emmanouil Valsomatzis, Katja Hose, and Torben Bach Pedersen. 2015. Aggregating and disaggregating flexibility objects. *IEEE Transactions on Knowledge and Data Engineering* (2015), 2893–2906.
- [33] Tea Tušar, Laurynas Šikšnys, Torben Bach Pedersen, Erik Dovgan, and Bogdan Filipić. 2012. Using aggregation to improve the scheduling of flexible energy offers. *Proc. of BIOMA* (2012), 347–358.
- [34] Emmanouil Valsomatzis, Katja Hose, and Torben Bach Pedersen. 2014. Balancing energy flexibilities through aggregation. In *DARE'2014*. 17–37.
- [35] Emmanouil Valsomatzis, Katja Hose, Torben Bach Pedersen, and Laurynas Šikšnys. 2015. Measuring and Comparing Energy Flexibilities.. In *EDBT/ICDT Workshops*. 78–85.
- [36] Emmanouil Valsomatzis, Torben Bach Pedersen, and Alberto Abelló. 2018. Day-ahead Trading of Aggregated Energy Flexibility (*e-Energy '18*). 134–138.
- [37] Emmanouil Valsomatzis, Torben Bach Pedersen, Alberto Abelló, and Katja Hose. 2016. Aggregating energy flexibilities under constraints. In *Smart Grid Communications (SmartGridComm), 2016 IEEE International Conference on*. 484–490.
- [38] Y. Vardanyan, F. Banis, S. A. Pourmousavi, and H. Madsen. 2018. Optimal co-ordinated bidding of a profit-maximizing EV aggregator under uncertainty. In *ENERGYCON*. 1–6.
- [39] Laurynas Šikšnys and Torben Bach Pedersen. 2016. Dependency-based FlexOffers: Scalable Management of Flexible Loads with Dependencies. In *Proceedings of the Seventh International Conference on Future Energy Systems (e-Energy '16)*. 1–13.
- [40] Rehman Zafar, Anzar Mahmood, Sohail Razzaq, Wamiq Ali, Usman Naeem, and Khurram Shehzad. 2018. Prosumer based energy management and sharing in smart grid. *Renewable and Sustainable Energy Reviews* (2018), 1675–1684.